JJM
SYSTEMS INC

C931020-U-2R05

AN EXPERT SYSTEM SHELL FOR INFERRING VEGETATION
CHARACTERISTICS - CHANGES TO THE
HISTORICAL COVER TYPE DATABASE (TASK F)

26 May 1993

Prepared for:

National Aeronautics and Space Administration
Goddard Space Flight Center
Greenbelt, MD 20771

Prepared by:

JJM Systems, Inc.
1225 Jefferson Davis Hwy., Suite 412
Arlington, VA 22202

JJM
SYSTEMS INC

TABLE OF CONTENTS

LIST OF FIGURES

JJM
SYSTEMS INC

LIST OF TABLES

LIST OF ACRONYMS

KEE   Knowledge Engineering Environment

VEG   VEGetation Workbench

SYSTEMS INC

SECTION 1.0

INTRODUCTION

All the options in the NASA VEGetation Workbench (VEG) make use of a database of historical cover types. This database contains results from experiments by scientists on a wide variety of different cover types. The learning system uses the database to provide positive and negative training examples of classes that enable it to learn distinguishing features between classes of vegetation. All the other VEG options use the database to estimate the error bounds involved in the results obtained when various analysis techniques are applied to the sample of cover type data that is being studied.

In the previous version of VEG, the historical cover type database was stored as part of the VEG knowledge base. This database has been removed from the knowledge base. It is now stored as a series of flat files that are external to VEG. An interface between VEG and these files has been provided. The interface allows the user to select which files of historical data to use. The files are then read, and the data are stored in Knowledge Engineering Environment (KEE) units using the same organization of units as in the previous version of VEG. The interface also allows the user to delete some or all of the historical database units from VEG and load new historical data from a file.

This report summarizes the use of the historical cover type database in VEG. It then describes the new interface to the files containing the historical data. It describes minor changes that were made to VEG to enable the externally stored database to be used. Test runs to test the operation of the new interface and also to test the operation of VEG using historical data loaded from external files are also described.

Task F has been completed. A Sun cartridge tape containing the KEE and Common Lisp code for the new interface and the modified version of the VEG knowledge base has been delivered to the NASA GSFC technical representative.

SECTION 2.0

BACKGROUND

The need for the historical cover type database in VEG is summarized in this section. Then, the organization of units in the historical cover type database in the VEG knowledge base is described.

2.1 THE USE OF THE HISTORICAL COVER TYPE DATABASE

All the options in VEG make use of the historical cover type database. This database contains results from experiments by scientist on a wide variety of different cover types.

When the learning system is in use, VEG selects positive and negative training examples from the historical cover type database. From these training examples, VEG determines relationships that discriminate between classes of vegetation. These classes can then used to classify an unknown sample. The learning system was described in detail in Kimes, Harrison and Harrison (1992) and JJM Systems report B921014-U-2R03. The historical cover type database is a fundamental part of the learning system.

All the other options in VEG use the historical cover type database in order to estimate the error terms when various analysis techniques are applied to the sample of cover type data being studied. A subset of historical data, referred to as the "restricted data set," is selected for each run. VEG can automatically select the restricted data set that best matches the sample. Alternatively, the user can indicate the bounds on each parameter of interest in and instruct VEG to select the subset of the historical cover type data that falls within the set bounds. Once the restricted data set has been selected, the reflectance data in each cover type are interpolated and extrapolated so that they match the exact view angles of the input spectral data. The restricted data set contains the true results, for example the spectral hemispherical reflectance, for each cover type. Each technique that is applied to the sample that is being studied is also applied to each cover type in the restricted data set. A difference score based on errors (true results minus calculated result) for the restricted data set is calculated for each technique. This score provides an estimate of the error involved in applying the technique to the sample being studied. The use of the historical cover type database was described in detail in Kimes, Harrison and Ratcliffe (1991) and JJM Systems report B921019-U-2R04.

2.2 ORGANIZATION OF HISTORICAL COVER TYPE DATABASE UNITS

The historical cover type database in the previous version of VEG was stored permanently in units in the VEG knowledge base. These units were subclasses and instances of the unit HISTORICAL.COVER.TYPES. Figure 2-1 shows part of this hierarchy of units. All the units that were children of the unit HISTORICAL.COVER.TYPES inherited slots from the parent unit. Figure 2-2 shows the slots in the unit HISTORICAL.COVER.TYPES.

Units such as COVER.TYPE.1 held data that were common for all the data sets for the particular cover type. These data were inherited by all the units such as CT1-26 and CT1-26-1 that were descendants of the unit COVER.TYPE.1. Data shared by all data sets for a particular cover type included the description, cover type, solar azimuth, solar zenith angles and the zenith interval. The DESCRIPTION slot of COVER.TYPE.1 contained the description "PLOWED FIELD - TUNISIA AFRICA - KIMES DATA SET." This description referred to all data sets collected at
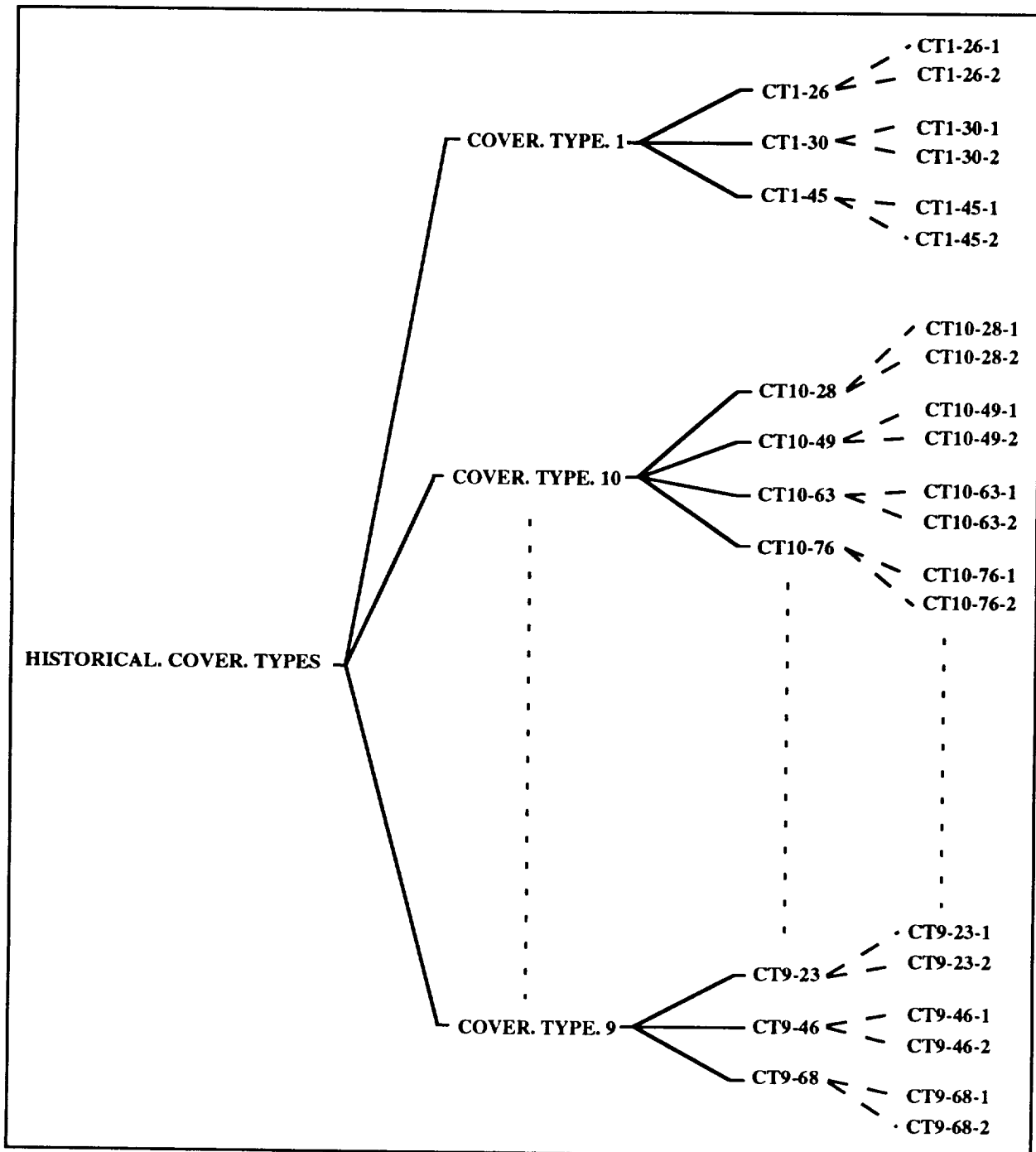
Figure 2-1
Some of the Units in the Historical Cover Type Database
in the Previous Version of VEG

```
AZIMUTH.INTERVAL
COVER.TYPE
DATE
DESCRIPTION
DRY.BIOMASS.KG.HC
GROUND.COVER
HEIGHT.CM
LEAF.AREA.INDEX
LEAF.ORIENTATION.DISTRIBUTION
LOCAL.STANDARD.TIME
MAX.ZENITH.DATA
PROPORTION.GREEN
RAW.DATA
SOLAR.AZIMUTH
SOLAR.ZENITH.ANGLE
SOLAR.ZENITH.ANGLES
SPECTRAL.HEM.REFLECTANCE
STRUCTURE
TOTAL.HEM.REFLECTANCE
WAVELENGTH.MAX
WAVELENGTH.MIN
WAVELENGTHS
WET.BIOMASS.KG.HC
ZENITH.INTERVAL
```

Figure 2-2
Slots in the Unit HISTORICAL.COVER.TYPES

the particular location. The SOLAR.AZIMUTH slot contained the value 180. The slot SOLAR.ZENITH.ANGLES held the list of solar zenith angles for which data were available for the cover type. The ZENITH.INTERVAL slot contained the value 15 indicating that measurements of reflectance for COVER.TYPE.1 were collected at 15 degree zenith intervals in each set of data.

The units such as CT1-26 and CT1-39 held data for COVER.TYPE.1 that were collected at solar zenith angles 27° and 39°, respectively. The data stored in these units were collected at the same location but at different times of the day when the sun was at different positions in the sky. The slot SOLAR.ZENITH.ANGLE in the units, such as CT1-26 and CT1-39, held the value of the solar zenith angle at the time that the data were collected. At each solar zenith angle, data were collected in one or more different wavebands.

The units CT1-26-1 and CT1-26-2 were instances of the unit CT1-26. These units held data collected at solar zenith angle 26° and in the wavebands 0.58 - 0.68 μm and 0.73 - 1.1 μm, respectively. The maximum and minimum wavelength in the wavebands were stored in the slots WAVELENGTH.MAX and WAVELENGTH.MIN. The spectral hemispherical reflectance measured in a particular waveband was stored in the slot SPECTRAL.HEM.REFLECTANCE. Reflectance data were stored in the slot RAW.DATA. The reflectance data consisted of reflectance measurements taken at different zenith and azimuth combinations. Each data point was specified by a zenith, azimuth and reflectance value. The data were stored in the slot RAW.DATA as a list of lists; e.g. ((0 0 0.231) (15 0 0.1968) (15 45 0.2094)). In this example, data at the nadir

(zenith and azimuth both zero) and at two other points were recorded. The reflectance values at the three points were 0.231, 0.1968 and 0.2094, respectively.

SYSTEMS INC

SECTION 3.0

CHANGES TO VEG

This section describes the interface that allows the user to load historical cover type data from external files and subsequently delete the loaded data from VEG. The format of the files that hold the historical data, and the structure of the units that hold the database when it is loaded into VEG are also described. The removal of the historical database from within the VEG hierarchy of units to a series of external files necessitated various minor changes to the operation of VEG and to the VEG user interface. These changes are described in this section.

3.1 CHANGING THE HISTORICAL COVER TYPE DATABASE

In the previous version of VEG when the user left-clicked on RUN.VEG, VEG was run and the Processing Mode Screen was displayed. This screen enabled the user to specify whether VEG should be run in the Automatic or Research mode. An additional option, "ADMINISTRATION," has been included in the menu as shown in Figure 3-1. This option allows the user to make various changes to VEG before processing the data.
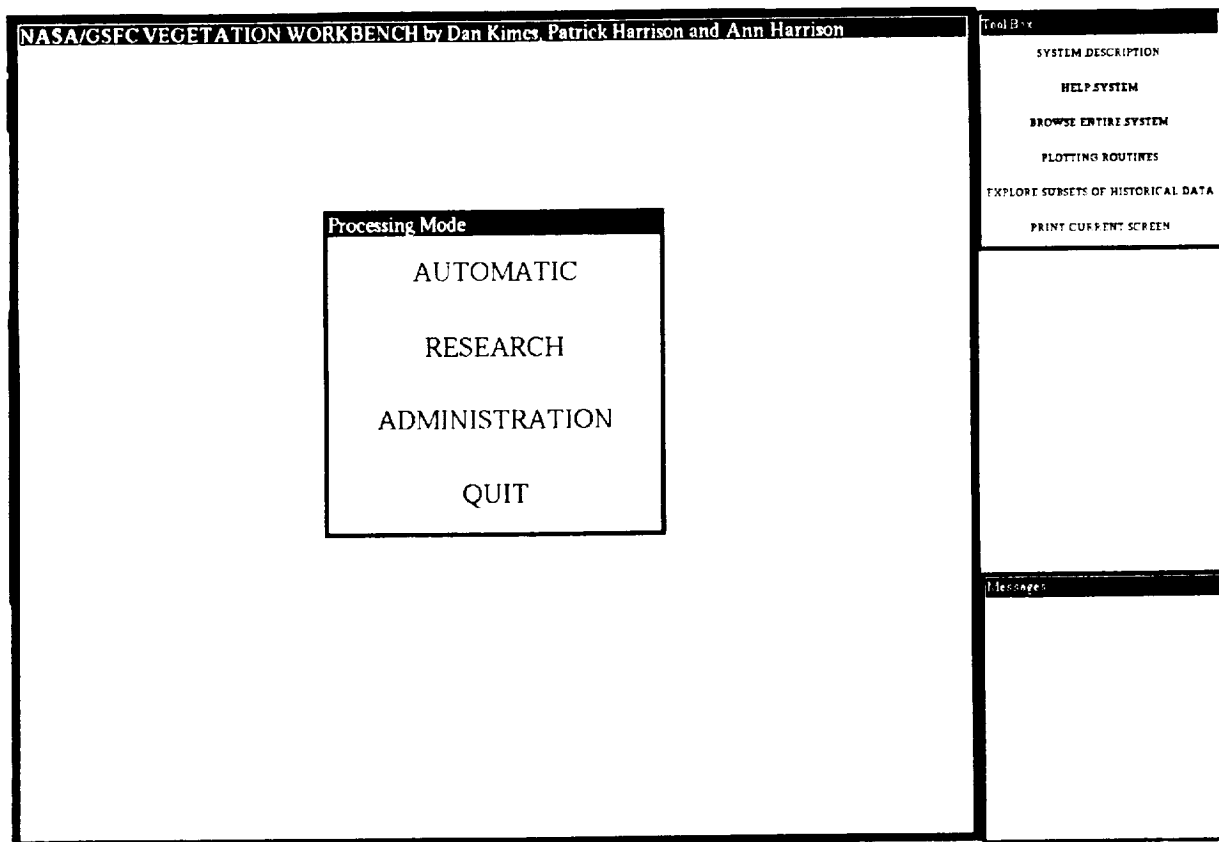
NASA/GSFC VEGETATION WORKBENCH by Dan Kimes, Patrick Harrison and Ann Harrison

Tool Box

SYSTEM DESCRIPTION

HELP SYSTEM

BROWSE ENTIRE SYSTEM

PLOTTING ROUTINES

EXPLORE SUBSETS OF HISTORICAL DATA

PRINT CURRENT SCREEN

Processing Mode

AUTOMATIC

RESEARCH

ADMINISTRATION

QUIT

Messages

Figure 3-1
The Processing Mode Screen in the Current Version of VEG

If the user selects the ADMINISTRATION option from the Processing Mode menu, the screen shown in Figure 3-2 is opened. This screen enables the user to either change the historical database or add techniques. The CHANGE.HISTORICAL.DATABASE option is described in this section of this report. The ADD.TECHNIQUES option was implemented as Task H of this contract, and it was described in JJM Systems Report C931021-U-2R07. More options may be added to this menu at a later date.
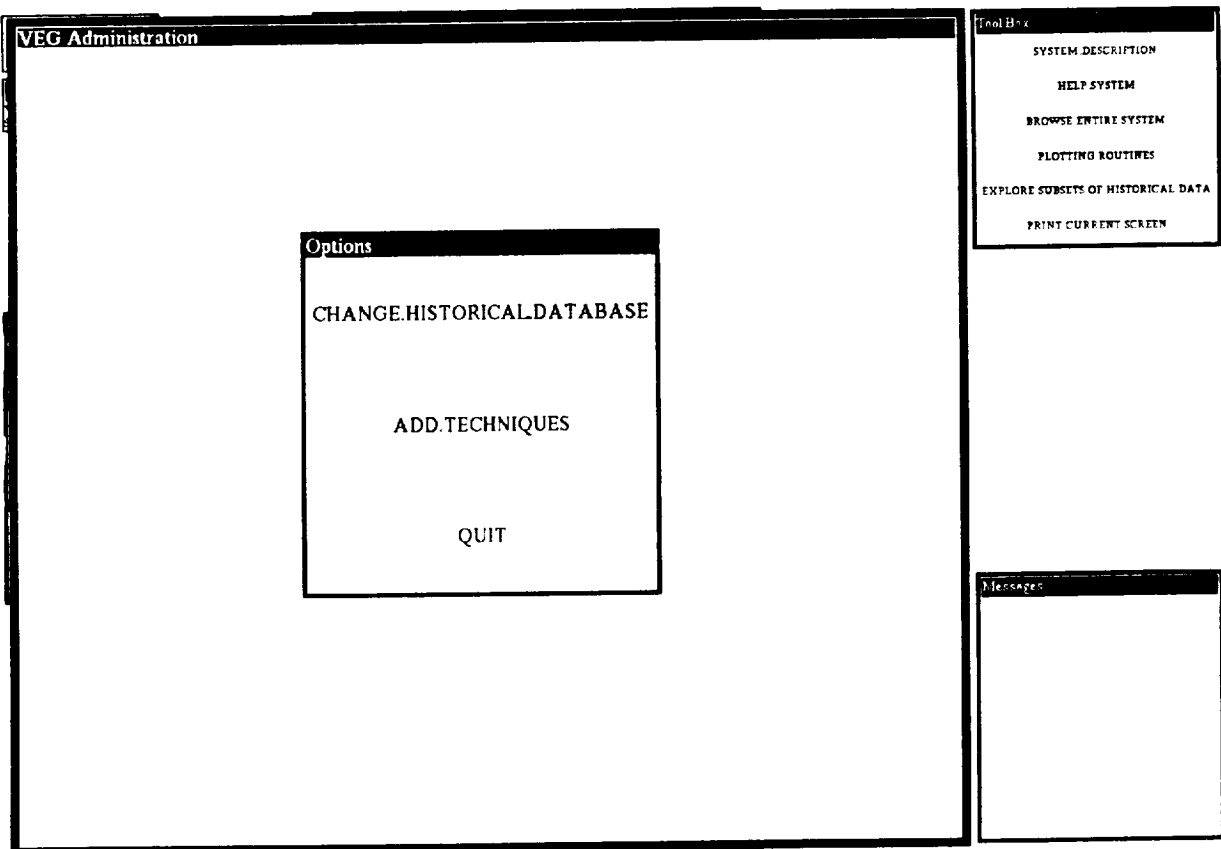

Figure 3-2
The Administration Screen

Left-clicking on the CHANGE.HISTORICAL.DATABASE option of the ADMINISTRATION menu, reveals the Change Historical Database Screen as shown in Figure 3-3. This screen allows the user to load historical data from an external file or delete previously loaded historical database units.

The unit CHANGE.HISTORICAL.DATABASE has been created in VEG. The subwindows (KEE ActiveImages) in the Change Historical Database Screen are attached to slots in this unit. The unit also holds the slots required by the methods that are used to add and remove cover types. For example, the names of the loaded databases are stored in the slot LOADED.DATABASES in this unit.

The database files are stored in the "historical-data" subdirectory of the directory containing the VEG files. All the database file names are in upper case letters. Each time the Change Historical Database Screen is opened, the names of the available databases are stored in the VALUECLASS facet of the AVAILABLE.DATABASES slot of the unit CHANGE. HISTORICAL.DATABASE. This enables the names of the available databases to be displayed using a "Vertical Pushbutton" KEE ActiveImage attached to the slot so the user can select a database by simply left-clicking on the database name.
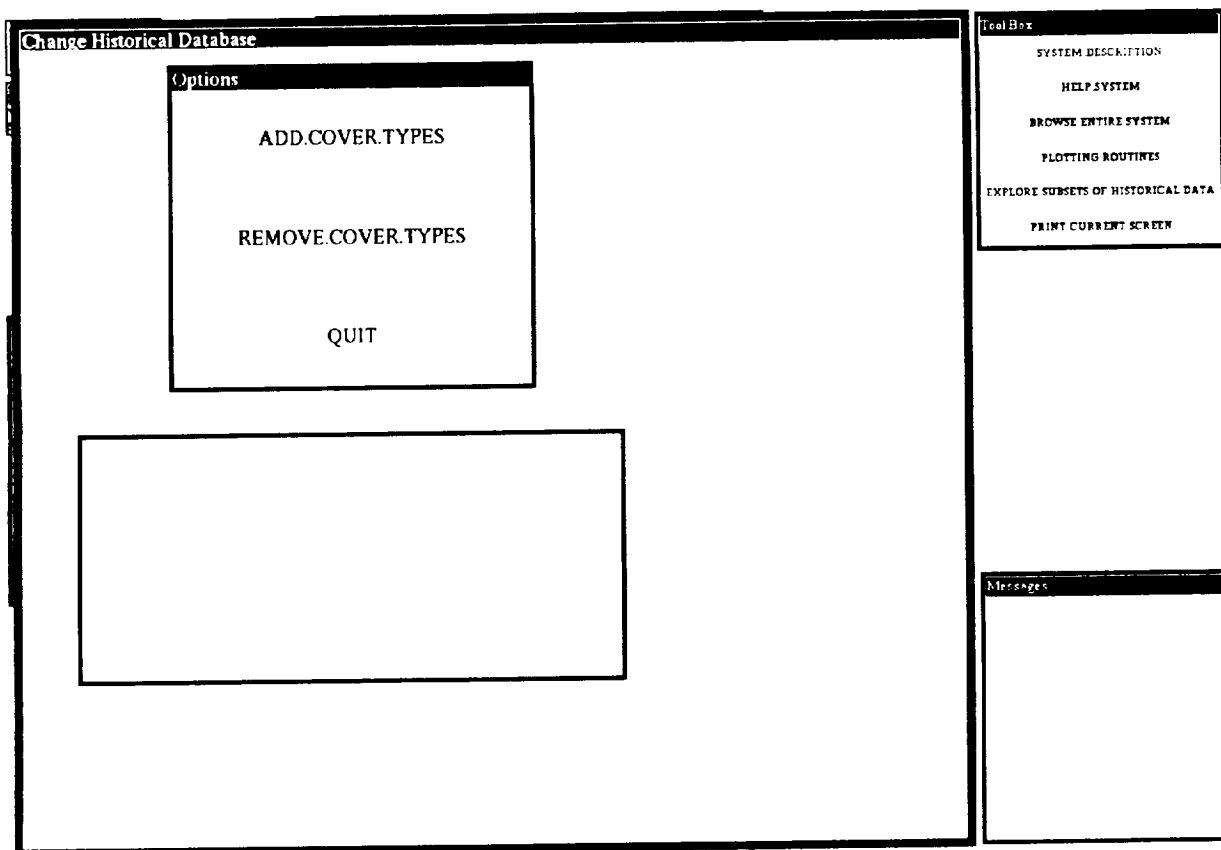

Figure 3-3
The Change Historical Database Screen

If the user selects the ADD.COVER.TYPES option (as shown in Figure 3-4) from the Change Historical Database Screen, the names of any databases that have already been loaded are displayed on the screen. A subwindow that is a Vertical Pushbutton ActiveImage attached to the AVAILABLE.DATABASES slot is opened. This subwindow contains the names of the available databases. The user is prompted to select the name of the database to load by left-clicking on the appropriate option in this subwindow.

VEG cannot run without a historical database. If the user attempts to run VEG before loading historical data, the Change Historical Database Screen is automatically opened. In this case, the ADD.COVER.TYPES option is automatically selected, indicating that the user should load historical data. The screen depicted in Figure 3-4 shows this situation.
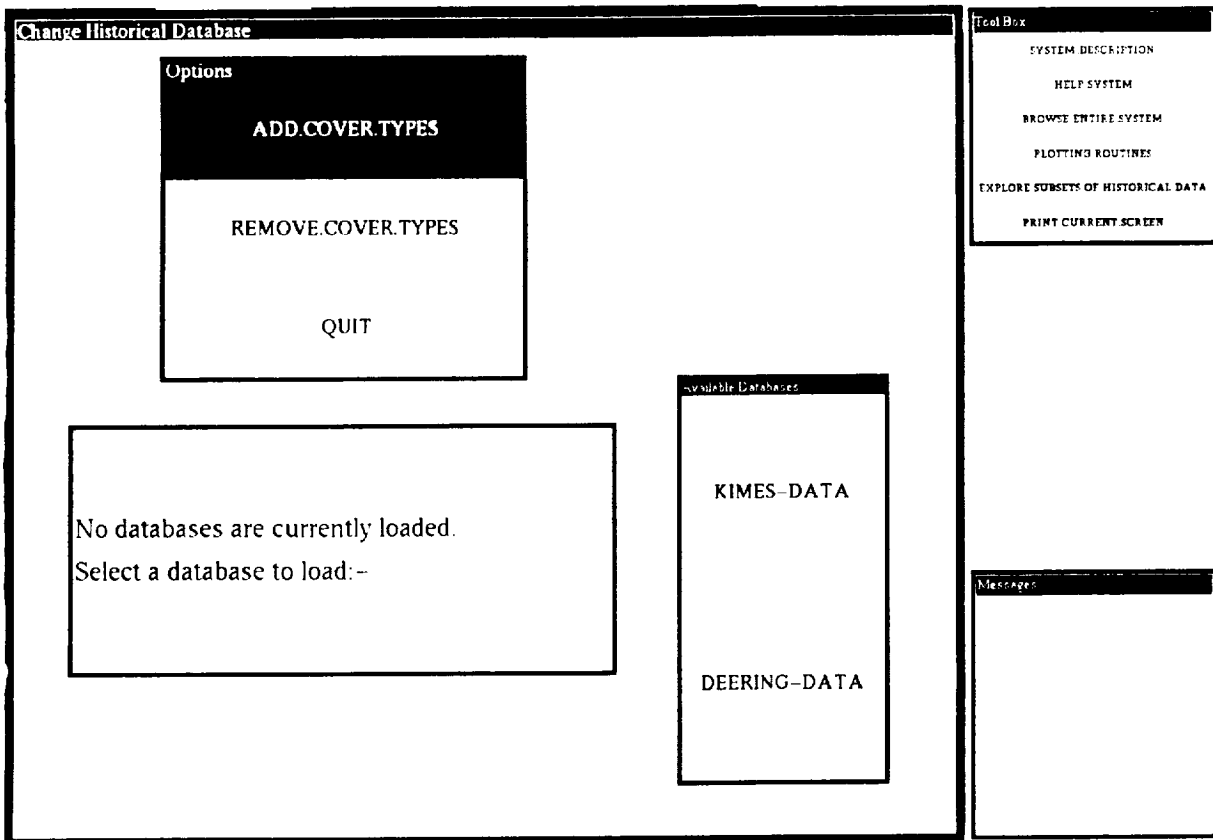
Change Historical Database

Options

ADD.COVER.TYPES

REMOVE.COVER.TYPES

QUIT

No databases are currently loaded.
Select a database to load:-

Available Databases

KIMES-DATA

DEERING-DATA

Tool Box

SYSTEM DESCRIPTION

HELP SYSTEM

BROWSE ENTIRE SYSTEM

PLOTTING ROUTINES

EXPLORE SUBSETS OF HISTORICAL DATA

PRINT CURRENT SCREEN

Messages

Figure 3-4
The Add Cover Types Option Before Any Cover Types have been Added

JJM
SYSTEMS INC

In the example shown in Figure 3-5, the database KIMES-DATA has already been loaded, and the user has selected the database "DEERING-DATA." The message "Loading ...." in the "Messages" box indicates that the data are being loaded from the file. When the loading has been completed, this message is removed. The newly loaded database is added to the list of loaded databases stored in the unit CHANGE.HISTORICAL.DATABASE. Its name is also displayed on the screen. If the user attempts to load data from a database that has already been loaded, a message is displayed and the database is not re-loaded. A message is also displayed if the user attempts to load data from an empty or missing database.

The format of the cover type data files, and an example of typical values are shown in Table 3-1. Data that apply to all data sets for the same cover type are stored first. These are followed by the data at each wavelength for the first solar zenith angle. The data for each wavelength for the remaining solar zenith angles are then listed. For clarity, only part of each set of reflectance data is shown in the table. In this example, data are available at solar zenith angles 26°, 30° and 45°. At each solar zenith angle, data are available in the wavebands 0.58 - 0.68 μm and 0.73 - 1.1 μm. In this example, the reflectance data beginning (0 0 0.231) and the spectral hemispherical reflectance value 0.1892 correspond to the solar zenith angle 26° and the waveband 0.58 - 0.68 μm. The reflectance data beginning (0 0 0.2733), and the spectral hemispherical reflectance value 0.2268 correspond to the solar zenith angle 26° and the waveband 0.73 - 1.1 μm

Change Historical Database

Options

ADD.COVER.TYPES

REMOVE.COVER.TYPES

QUIT

Currently loaded databases: KIMES DATA
Select a database to add: -

Tool Box

SYSTEM DESCRIPTION

HELP SYSTEM

BROWSE ENTIRE SYSTEM

PLOTTING ROUTINES

EXPLORE SUBSETS OF HISTORICAL DATA

PRINT CURRENT SCREEN

Available Databases

KIMES-DATA

DEERING-DATA

Messages

Loading ....

Figure 3-5
Adding Cover Types

Table 3-1
Historical Data File Format and an Example of Typical Values

TYPICAL VALUES	DESCRIPTION
"PLOWED-FIELD - TUNISIA AFRICA"	Description
45	Azimuth Interval
SOIL	Cover Type
(4 28 1983)	Date
0.0	Dry Biomass
0.0	Ground Cover
0.0	Height
0.0	Leaf Area Index
UNKNOWN	Leaf Orientation Distribution
(0909 1045 1217)	Local Standard Times
75	Maximum Zenith Data
0.0	Proportion Green
180	Solar Azimuth
HOMOGENEOUS	Structure
0.0	Total Hem. Reflectance
((0.58 0.68) (0.73 1.1))	Wavelengths Available
0.0	Wet Biomass
15	Zenith Interval
(26 30 45)	Solar Zenith Angles
((0 0 0.231) (15 0 0.1968) ... (75 315 0.1115))	Reflectance Data
0.1892	Spectral Hem. Reflectance
((0 0 0.2733) (15 0 0.2362) ... (75 315 0.137))	Reflectance Data
0.2268	Spectral Hem. Reflectance
((0 0 0.204) (15 0 0.1599) ... (75 315 0.101))	Reflectance Data
0.1813	Spectral Hem. Reflectance
((0 0 0.244) (15 0 0.1921) ... (75 315 0.1262))	Reflectance Data
0.2173	Spectral Hem. Reflectance
((0 0 0.1738) (15 0 0.1451) ... (75 315 0.0923))	Reflectance Data
0.1868	Spectral Hem. Reflectance
((0 0 0.2033) (15 0 0.1691) ... (75 315 0.1103))	Reflectance Data
0.2219	Spectral Hem. Reflectance

SYSTEMS INC

When data for a cover type are read from a file, a hierarchy of KEE units is created to hold the data. Data which are common to all sets of data for the same cover type are stored in a unit which is created as a subclass of the unit HISTORICAL.COVER.TYPES. The name of this unit begins with "COVER.TYPE." This name is automatically generated. The name of the data file from which the cover type data has been read is also stored in the cover type unit. The purpose of this is to identify the data file that was the source of the data so that all data from a selected data file can later be identified prior to deletion. A subclass of the cover type unit is created for each solar zenith angle for which data are available. Instances of these units are created to hold the reflectance data, spectral hemispherical reflectance value and the maximum and minimum wavelengths in each waveband. This is the same organization of units that was used in the previous version of VEG. As the data are read from the file, the required units are created and data are stored in the units. Each data value is checked before it is stored. If the value is of the wrong type, or it is out of range for the slot in which it is to be stored, an error message is displayed on the screen. If an error is detected, processing of the data file is aborted and all the data for the cover type that is currently being stored are deleted from VEG. Any cover types that have been previously stored correctly are not deleted. Figure 3-6 shows a hierarchy of KEE units that might be created to hold the data shown in Table 3-1.
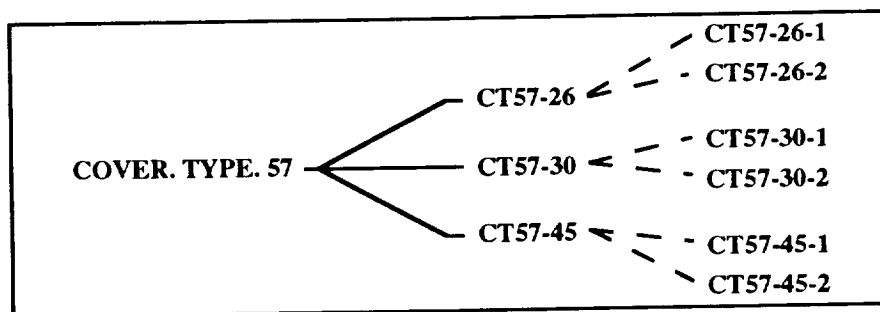
```
                                                      ╱ CT57-26-1
                                          ╭ CT57-26 ⪦    CT57-26-2

COVER. TYPE. 57 ⪦──────── CT57-30 ⪦       ─ CT57-30-1
                                                     ─ CT57-30-2

                          ╰ CT57-45 ⪦    ─ CT57-45-1
                                              ╲ CT57-45-2
```

Figure 3-6
KEE Units that Hold the Data from Table 3-1

If the user selects the REMOVE.COVER.TYPES option from the Change Historical Database menu, a different subwindow is opened and the names of the currently loaded databases are displayed on the screen, as shown in Figure 3-7. The user is prompted to select the database to be removed or to select "ALL" if all the databases are to be removed. If the user selects "ALL," all the units that are descendants of the unit HISTORICAL.COVER.TYPES are deleted from VEG, and all the values are removed from the LOADED.DATABASES slot of the unit CHANGE.HISTORICAL.DATA. The message on the screen is then updated to indicate that no databases are currently loaded, and the subwindow showing the loaded databases is removed from the screen. Alternatively, if the user selects a database to be removed, VEG searches through all the cover type units that are subclasses of the unit HISTORICAL.COVER.TYPES and identifies the cover types that originated from the selected database file. These cover type units and all their subclass and member units are then deleted. The name of the deleted database is removed from the LOADED.DATABASES slot of the unit CHANGE.HISTORICAL.DATA and the subwindow containing the names of the loaded databases is updated on the screen. If no databases are loaded, the subwindow containing the names of the loaded databases is removed from the screen. Note that when databases are removed from VEG, the underlying files are not modified or destroyed.

Selecting QUIT from the Change Historical Database menu closes this screen. If the user previously selected the Change Historical Database option from the Administration menu, the

JJM SYSTEMS INC

Administration menu is again displayed. If the Change Historical Database Screen was opened automatically by VEG when either the automatic or the research mode was selected because no historical data was present, the Processing Mode Screen (Figure 3-1) is reopened.

Change Historical Database

Options

ADD.COVER.TYPES

REMOVE.COVER.TYPES

QUIT

Select the name of the database to remove, or ALL to remove all the databases:-

Database to Remove

ALL

DEERING-DATA

KIMES-DATA

Tool Box

SYSTEM DESCRIPTION

HELP SYSTEM

BROWSE ENTIRE SYSTEM

PLOTTING ROUTINES

EXPLORE SUBSETS OF HISTORICAL DATA

PRINT CURRENT SCREEN

Messages

Figure 3-7
Removing Cover Types

JJM
SYSTEMS INC

3.2 CHANGES TO THE PICK SUBSET SCREEN

The names of the historical cover type database units were hard-wired into various slots in the previous version of VEG. Changes have been made to these slots and the screens that use them to provide the greater flexibility required in the current version of VEG.

In all the VEG options except the learning system, the user can choose to select the restricted data set manually. If the manual Pick Restricted Data Set option is selected, the Pick Restricted Data Set Screen is opened. This screen allows the user to enter the maximum and minimum values to be considered for parameters such as height and solar zenith angle. The historical cover type database is then searched to find the cover types that match the criteria entered by the user. The user can select a subset of the matched data using the Pick Subset Screen shown in Figure 3-8. In the previous version of this screen, the user selected a cover type by left-clicking on a choice in a menu of available cover types that were hard-wired into a slot in VEG. In order to provide the additional flexibility required by the current version of VEG, the subwindow (ActiveImage) labeled "Current Cover Type" is updated to show the names of currently available cover types each time the Pick Subset Screen is opened. This is done by changing the VALUECLASS facet of the slot to which the ActiveImage is attached. The operation of the Pick Subset Screen is the same as in the previous version of VEG.

Figure 3-8
The Current Version of the Pick Subset Screen

3.3 CHANGES TO THE BROWSER

In the previous version of VEG, the names of the historical cover type units were hard-wired into a slot that was displayed on the Cover Type Descriptions Screen of the Browser (shown in Figure 3-9). This screen can be selected by choosing the EXPLORE.SUBSETS.OF.HISTORICAL.DATA browser option and then picking the SEE.DESCRIPTIONS option. The Lisp code controlling the Cover Type Descriptions Screen has been modified. Each time the Cover Type Descriptions Screen is opened, the names of the currently available cover types are stored in the VALUECLASS facet of the slot CT.TO.USE of the unit DATA.MATCHER. This causes the updating of the Vertical Pushbutton ActiveImage (subwindow) labeled "Cover Type" in Figure 3-9 which is attached to the slot. As in the previous version of VEG, when the user left-clicks on a cover type in the "Cover Type" subwindow, the description of the cover type is displayed in the "Description" subwindow. If no cover types are available, the message "No cover types are currently available" is displayed in the "Description" subwindow and the "Cover Type" subwindow is removed from the screen. The screen is closed by selecting the QUIT option at the foot of the screen.

The Cover Type Descriptions Screen may be open at the same time as the Change Historical Database Screen. In this case, the Cover Type Descriptions Screen is updated every time cover types are added to or removed from the historical cover type database.
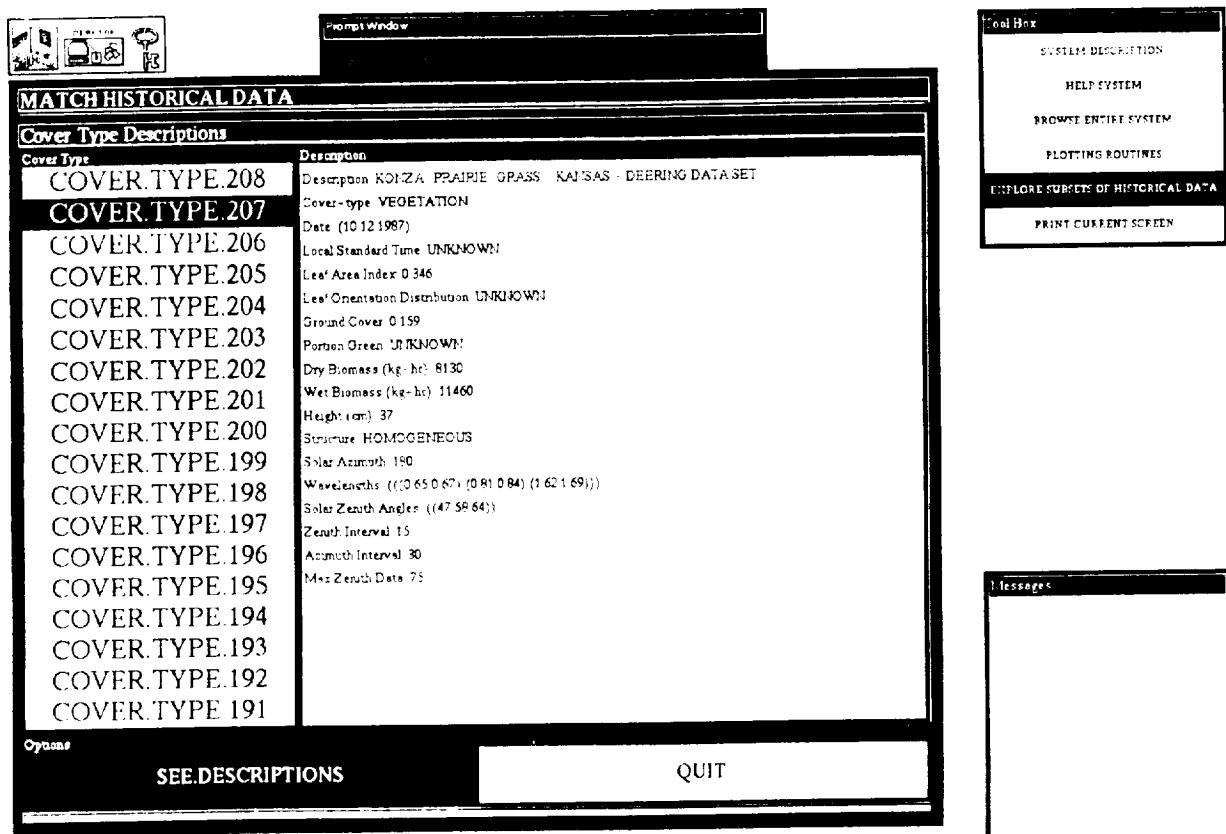

Figure 3-9
The Cover Type Description Screen in the Browser

SYSTEMS INC

SECTION 4.0

TESTING AND RESULTS

All the Change Historical Database options were tested using both valid and invalid inputs. All the VEG goals, including the learning system, were tested using historical cover type data loaded from external files. After some minor changes necessitated because the available wavelength data were stored in a different format than in the previous version of VEG, all the options were found to be operating correctly. The test runs are described in detail in this section.

3.1   TEST 1

The purpose of Test 1 was to test the navigation back and forth through the various menu levels from the VEG top level to the Change Historical Database Screen. The user left-clicked on RUN.VEG, ADMINISTRATION and CHANGE.HISTORICAL.DATABASE on successive screens. As expected, the Change Historical Database Screen was opened. The user then selected QUIT in each successive menu to navigate back to the top level of VEG. This test showed that the screens between the VEG top level and the Change Historical Database Screen were opened and closed in the correct sequence.

3.2   TEST 2

This test was designed to test the ADD.COVER.TYPES option of the Change Historical Database menu. This option was selected, and the user was prompted to select the database to add to the historical cover type database. According to the display, the databases KIMES-DATA and DEERING-DATA were available. The user selected "KIMES-DATA." The message "Loading ...." appeared in the "Messages" box while the database was being loaded. Inspection of the KEE knowledge base confirmed that the database had been correctly loaded.

The user then attempted to load the same database again by again selecting "KIMES-DATA" as the database to be loaded. The message "Database KIMES-DATA has already been loaded" was displayed in the "Messages" box. The database was not reloaded.

The next part of this test, was designed to confirm that a second database could be loaded. The database "DEERING-DATA" was selected. The DEERING-DATA database was successfully loaded in addition to the KIMES-DATA database.

The operation of the ADD.COVER.TYPES option with invalid databases was tested in the final part of this test. Three new files were added to the historical-data subdirectory. These were named "EMPTY," "MISSING" and "FAULTY-DATA." The file "EMPTY" was empty. The files "MISSING" and "FAULTY-DATA" were copies of the file "KIMES-DATA." Several data items that were part of the second cover type in the file "FAULTY-DATA" were deleted so that the file had an incorrect format. The ADD.COVER.TYPES option was deselected and selected again. The names of the newly created files were then shown in the "Available Databases" subwindow. The user selected the "EMPTY" database. The message "Database file is empty" was displayed in the "Messages" box. The user then deleted the file "MISSING" from the historical-data subdirectory. When the user selected the "MISSING" file in the Change Historical Database screen, the message "Database file not found" was displayed in the "Messages" box. Finally, the user selected the "FAULTY-DATA" file. The first cover type was successfully loaded from this file. The message "File reading aborted - the data (843 1053 1254) is invalid for the slot

SYSTEMS INC

LEAF.AREA.INDEX." was then displayed in the "Messages" box and the reading of the file was aborted.

Test 1 showed that the ADD.COVER.TYPES option was working correctly with both valid and invalid databases.

3.3 TEST 3

The REMOVE.COVER.TYPES option was thoroughly tested in Test 3. At the beginning of this test, both databases were loaded, as in the previous test. Then the REMOVE.COVER.TYPES option was selected. The prompt confirmed that both databases were loaded and asked the user to select the database to remove, or "ALL" if all databases were to be removed. The user selected "KIMES-DATA." The database KIMES-DATA was removed. Inspection of the VEG knowledge base confirmed this. The content of the "Database to Remove" subwindow changed to report that the only database now loaded was DEERING-DATA.

The user then specified that the DEERING-DATA database should be deleted. After the deletion, the prompt on the screen indicated that no databases were currently loaded and the "Database to Remove" subwindow was removed from the screen.

The final part of this test was designed to test the option that causes all loaded databases to be removed. First, both databases were again loaded. The option REMOVE.COVER.TYPES was then selected again, and the user selected "ALL." As expected, all the databases were deleted, the prompt on the screen then reported that no databases were loaded, and the "Database to Remove" subwindow was removed from the screen.

Test 3 confirmed that all parts of the REMOVE.COVER.TYPES option were operating correctly.

3.4 TEST 4

Test 4 was designed to test the operation of VEG using historical data loaded from external files. At the beginning of this test, the historical cover type database was empty. The user left-clicked on RUN.VEG and then RESEARCH. The Change Historical Database Screen was automatically opened and the ADD.COVER.TYPES option was automatically selected. The databases KIMES-DATA and DEERING-DATA were then loaded. When the user selected QUIT, the Processing Mode menu was again revealed. The user left-clicked on RESEARCH again. This time, the Goals menu was displayed and the user selected the goal SPECTRAL.HEMISPHERICAL.REFLECTANCE. This option was run using SAMPLE3 from the VEG knowledge base.

In this test run, the user chose to manually select the restricted data set. Using the User Pick Restricted Data Set Screen, the user indicated that cover types with the same wavelengths as SAMPLE3, solar zenith angles between 45° and 60° and height less than 500 cm should be placed in the restricted data set. Figure 4-1 shows this screen after the cover types that match the chosen parameters had been identified. The user then decided to pick a subset of the matched data using the Pick Subset Screen. The sun angle 58° for the COVER.TYPE.27 was removed using this screen as shown in Figure 4-2.

JJM
SYSTEMS INC

Figure 4-1
The User Picks Restricted Data Set Screen After the Cover Types
that Match the Chosen Parameters have been Identified

JJM SYSTEMS INC

Figure 4-2
The Pick Subset Screen After a Cover Type has been Removed

SYSTEMS INC

The processing of the sample was then completed and the results were displayed as shown in Figure 4-3. This test showed that SPECTRAL.HEMISPHERICAL. REFLECTANCE operated correctly using historical cover type data loaded from external files. The test also showed that the Pick Subset Screen that had been modified to deal with the historical data loaded from files was operating correctly.

Figure 4-3
The Output Screen at the End of Test 4

4.5 TEST 5

In Test 5, VEG was run in Automatic Mode and the operation of all the VEG goals except the Learning System was tested. In Automatic mode, the restricted data set is selected automatically by the system. Several of the test runs from Task B used the VEG Automatic mode. When Task B was performed, the historical cover type database was stored in the VEG knowledge base. The test runs were reported in JJM Systems Report B921016-U-2R02. In Test 5 of the current task, databases KIMES-DATA and DEERING-DATA were loaded into VEG. The selected Task B tests were then repeated. The results of the current runs were compared with the Task B results. The results from the two sets of runs were found to be very similar but not identical. In particular, there were small but insignificant differences in the error terms and coefficients. The differences can be explained by the fact that the restricted data set is organized in a different order in the current version of VEG compared with the previous version. The algorithm that selects the restricted historical data picks the first ten statistical matches. For example, if the DEERING-DATA database is loaded before the KIMES-DATA database, the first ten matches may not be the same as in the case in which the KIMES-DATA database is loaded after the DEERING-DATA database. If the first ten matches are different, a different set of cover types are selected for calculating the coefficients and the error terms. This leads to slightly different results.

Test 5 showed that all the VEG goals tested were operating correctly using historical cover type data loaded from external files.

4.6 TEST 6

The Learning System was tested in Test 6 to confirm that it would operate correctly using historical cover type data loaded from external files. The Learning System was developed in Tasks C and D. When these tasks were tested, the historical database was part of the VEG knowledge base. In Test 6 of the current task, test runs 1, 2 and 3 from the Task C and D report, JJM Systems Report B921014-U-2R03, were repeated using the current version of VEG, including historical data loaded from external files. The results of Test 6 were identical to the results from the Tasks C and D in the earlier report. This provided further confirmation that the historical data loaded from external files had been correctly incorporated in VEG.

4.7 TEST 7

In this test the Cover Type Description Screen of the Browser was tested. The Lisp code controlling this screen had been modified so that it would work with historical data loaded from external files. This screen was selected by choosing the EXPLORE.SUBSETS.OF.HISTORICAL.DATA browser option and then picking the SEE.DESCRIPTIONS option. The display indicated that cover types 564 through 581 were available. The user selected COVER.TYPE.570. A description of this cover type was displayed.

The second part of this test was designed to test whether the Cover Type Description Screen would be correctly updated if cover types were added to or removed from the historical cover type database while the screen was open. Without closing the Cover Type Description Screen, the user opened the Change Historical Database Screen. The user then removed all the historical cover type data from VEG. Next, the user reloaded the KIMES-DATA database and then the DEERING-DATA database. Finally, the user removed the DEERING-DATA database. Inspection of the Cover Type Description Screen after each step in this test confirmed that it was being correctly updated.

JJM
SYSTEMS INC

Test 7 showed that the Cover Type Description Screen of the Browser was operating correctly.

SECTION 5.0

CONCLUSIONS

The historical cover type database has been removed from VEG and stored as a series of external files. An interface has been implemented. This interface allows the user to load historical cover type data from the files into VEG and subsequently delete the loaded data from VEG. Some minor changes were made to VEG to enable it to operate with the loaded data.

All the options provided by the new interface were tested. Data were loaded from external files into VEG and the operation of all the VEG goals was then tested. The test runs showed that the interface was working correctly and that the data had been loaded successfully from the external files into VEG. The tests also confirmed that the historical cover type data were correctly integrated into VEG.

The database of historical cover types in the previous version of VEG occupied 123 units. Removing this database to external files and only loading a subset of the data has reduced the memory requirements for VEG.

Allowing the user to select the historical data to load has made VEG more flexible. VEG is now useful to a wider group of scientists. Scientists with different areas of interest can use different sets of historical data with VEG and restrict the use of VEG to the particular types of data that are of interest to them.

Relational database environments could easily be used to hold the historical cover type data in place of the flat files. However, the cost of the KEE interface to a relational database is presently prohibitive.

JJM SYSTEMS INC

REFERENCES

JJM Systems, Inc. April 1993. An Expert System for Inferring Vegetation Characteristics - Interface for the Addition of Techniques (Task H). Ivyland, PA. C931021-U-2R07.

JJM Systems, Inc. October 1992. An Expert System for Inferring Vegetation Characteristics - Implementation of Additional Techniques (Task E). Ivyland, PA. B921019-U-2R04.

JJM Systems, Inc. September 1992. An Expert System for Inferring Vegetation Characteristics - The Learning System (Tasks C and D). Ivyland, PA. B921014-U-2R03.

JJM Systems, Inc. September 1992. An Expert System for Inferring Vegetation Characteristics - Output of Results to a File (Task B). Ivyland, PA. B921016-U-2R02.

Kimes, D.S., Harrison, P.R. and Harrison, P.A.. March 1992. Learning Class Descriptions from a Data Base of Spectral Reflectance with Multiple View Angles. In IEEE Transactions on Geoscience and Remote Sensing, Vol. 30, No 2, pp. 315-325.

Kimes, D.S., Harrison, P.R. and Ratcliffe, P.A.. October 1991. A Knowledge-Based Expert System for Inferring Vegetation Characteristics. International Journal of Remote Sensing, Vol 12, no 10: pp. 1987-2020.

JJM
SYSTEMS INC

APPENDIX A

LISTING OF CODE FOR CHANGING THE HISTORICAL DATABASE

JJM
SYSTEMS INC

```
;;;; veg-methods4.lisp
;;;
;;;
;;; Task F
;;; Methods for making historical cover type database external to VEG
;;;
;;;
;;; Written by Ann & Patrick Harrison
;;; Created March 16 1993
;;; Last Modified May 21 1993


(in-package `kee)

(defun input-historical-data-from-file (file db-name)
"Controls the input of historical cover type data from a file."
   (catch 'invalid-historical-data
      (my-documentation-print "Loading ........")
      (with-open-file (str file :direction :input)
        (store-historical-data db-name str)))
     (update-current-cover-types))


(defvar *historical-data-ct-slot-list*
    '(azimuth.interval cover.type date dry.biomass.kg.hc ground.cover
height.cm leaf.area.index leaf.orientation.distribution local.standard.time
max.zenith.data proportion.green solar.azimuth structure total.hem.reflectance
wavelengths wet.biomass.kg.hc zenith.interval solar.zenith.angles)
"Slots in which to store historical data at the cover type level.")


(defvar *historical-data-wavelength-slot-list*
    '(raw.data spectral.hem.reflectance)
"Slots in which to store historical data at the wavelength level.")


;;; Note that the function read-file is included in the methods file
;;; veg-methods1.lisp.


(defun store-historical-data (db-name str)
"Stores the data for any number of cover-types."
   (add.value 'change.historical.database 'loaded.databases db-name)
   (do ((first-slot (read-file str)(read-file str)))
       ((null first-slot)(clear-prompt))
      (let ((new-cover-type (create.unit
                        (gentemp "COVER.TYPE.")
                        'veg 'historical.cover.types ()))
           (wavelengths nil)
           (solar-zenith-angles nil))
       (put.value new-cover-type 'description first-slot)
       (put.value new-cover-type 'database db-name)
       (dolist (slot *historical-data-ct-slot-list*)
          (let ((data (read-file str)))
            (cond ((null data)
                    (abort-historical-data-reading new-cover-type 'eof))
                                       ;Eof in wrong place
                  ((not (valid-historical-ct-data data slot))
                   (abort-historical-data-reading new-cover-type data slot))
                  (t (put.value new-cover-type slot data)))
            (when (eq slot 'wavelengths)
```

JJM
SYSTEMS INC

```
          (setf wavelengths data))
        (when (eq slot 'solar.zenith.angles)
          (setf solar-zenith-angles data))))
      (process-sun-ang-data str new-cover-type solar-zenith-angles
                            wavelengths))))


(defun process-sun-ang-data (str new-cover-type solar-zenith-angles
                             wavelengths)
"Creates the required units and stores the sun angle data."
  (dolist (sun-ang solar-zenith-angles)
    (let ((new-sun-ang (create.unit
                         (get-sun-angle-unit-name new-cover-type sun-ang)
                         'veg new-cover-type)))
      (put.value new-sun-ang 'solar.zenith.angle sun-ang)
      (process-wave-data str new-cover-type new-sun-ang wavelengths))))


(defun process-wave-data (str new-cover-type new-sun-ang wavelengths)
"Creates the required units and stores the data at the wavelength level.  These
data consist of the maximum and minimum wavelengths, raw data and spectral
hemispherical reflectance."
  (let ((n 1))
    (dolist (wave wavelengths)
      (let ((new-wave (create.unit
                        (get-wave-unit-name new-sun-ang n)
                        'veg nil new-sun-ang)))
        (put.value new-wave 'wavelength.min (first wave))
        (put.value new-wave 'wavelength.max (second wave))
        (dolist (slot *historical-data-wavelength-slot-list*)
          (let ((data (read-file str)))
            (cond ((null data)
                    (abort-historical-data-reading new-cover-type 'eof))
                                       ;Eof in wrong place
                  ((not (valid-historical-wave-data data slot))
                    (abort-historical-data-reading new-cover-type data slot ))
                  (t (put.value new-wave slot data)))))
        (incf n)))))


(defun get-sun-angle-unit-name (new-cover-type new-sun-ang)
"Returns the name of a unit composed of the combination of the cover-type name
and the sun angle."
  (intern (format () "CT~A-~A"
                  (string-trim "COVER.TYPE."
                               (string (unit.name new-cover-type)))
                  new-sun-ang)))


(defun get-wave-unit-name (new-sun-ang n)
"Returns the name of a unit composed of the combination of the sun-angle unit
name and a number."
  (intern (format () "~A-~A"
                  (unit.name new-sun-ang)
                  n)))
```

JJM
SYSTEMS INC

```
(defun valid-historical-ct-data (data slot)
"Returns t if the data are valid for the slot and nil otherwise."
  (case slot
    (azimuth.interval (and (integerp data)
                           (>= data 0)
                           (<= data 45)))
    (cover.type (member data `(soil vegetation)))
    (date t)
    (dry.biomass.kg.hc (or (eq data 'unknown)
                           (and (numberp data)
                                (>= data 0)
                                (<= data 25000))))
    (ground.cover (or (eq data 'unknown)
                      (and (numberp data)
                           (>= data 0)
                           (<= data 1))))
    (leaf.orientation.distribution t)
    (local.standard.time t)
    (max.zenith.data (and (integerp data)
                          (>= data 0)
                          (<= data 90)))
    (proportion.green (or (eq data 'unknown)
                          (and (numberp data)
                               (>= data 0)
                               (<= data 1))))
    (solar.azimuth (and (integerp data)
                        (>= data 0)
                        (<= data 360)))
    (structure (member data '(homogeneous heterogeneous)))
    (total.hem.reflectance (and (numberp data)
                                (>= data 0)
                                (<= data 1)))
    (wavelengths (and (consp data)
                      (dolist (waves data t)
                        (unless (and (consp waves)
                                     (= (length waves) 2)
                                     (dolist (wave waves t)
                                       (unless (and (numberp wave)
                                                    (>= wave 0)
                                                    (<= wave 10))
                                         (return nil))))
                          (return-from valid-historical-ct-data nil)))))
    (wet.biomass.kg.hc (or (eq data 'unknown)
                           (and (numberp data)
                                (>= data 0)
                                (<= data 25000)))) ;Dan confirm
    (zenith.interval (and (integerp data)
                          (>= data 0)
                          (<= data 45)))
    (description (stringp data))
    (leaf.area.index (or (eq data 'unknown)
                         (and (numberp data)
                              (>= data 0)
                              (<= data 10))))
```

JJM
SYSTEMS INC

```lisp
(height.cm (and (numberp data)
                (>= data 0)
                (<= data 3000)))
(number.wavelengths (and (integerp data)
                         (>= data 0)
                         (<= data 10)))
(solar.zenith.angles (and (consp data)
                          (dolist (dat data t)
                             (unless (and (integerp dat)
                                          (>= dat 0)
                                          (<= dat 90))
                                (return-from valid-historical-ct-data nil)))))))

(defun valid-historical-wave-data (data slot)
"Returns t if the data are valid for the slot and nil otherwise."
  (case slot
    (raw.data (valid-reflectance-data data))
    (t (and (numberp data) ;Must be spectral hemispherical reflectance
            (>= data 0)
            (<= data 1)))))

(defun abort-historical-data-reading (new-cover-type data &optional slot)
"Displays an error message and aborts the reading of the file if invalid data
are encountered or the end of file is encountered in the wrong place."
  (put.value 'methods 'general.message
      (if (eq data 'eof)
          "File reading aborted - end of file encountered prematurely"
          (format ()
             "File reading aborted - the data ~S is invalid for the slot ~S"
             data slot)))
  ;;; Remove from the historical data base all data for this cover type."
  (dolist (sun-ang (unit.children new-cover-type 'subclass))
    (dolist (wave (unit.children sun-ang 'member))
      (delete.unit wave))
    (delete.unit sun-ang))
  (delete.unit new-cover-type)
  (throw 'invalid-historical-data nil))

(defun open-admin-menu ()
"Opens and initializes the Admin menu."
  (remove.all.values 'admin 'options)
  (unitmsg 'viewport-admin.1 'open-panel!))

(defun open-change-historical-database-menu ()
"Opens and initializes the Change Historical Database menu."
  (unitmsg 'viewport-historical.database.1 'open-panel!)
  (remove.all.values 'change.historical.database 'options)
  (put.value 'change.historical.database 'entry.box "")
  (put.value 'change.historical.database 'message ""))
```

JJM
SYSTEMS INC

```lisp
(defun empty-db-file (db)
"Returns t if the file is empty and nil otherwise."
   (with-open-file (str db :direction :input)
     (let ((len (file-length str)))
        (or (null len)
            (zerop len)))))

(defun correct-empty-db ()
"This function is called when the historical cover type database is found to be
empty. The Change Historical Database screen is opened and set to prompt the
user to add cover types."
   (unitmsg 'viewport-historical.database.1 'open-panel!)
   (put.value 'change.historical.database 'entry.box "")
   (put.value 'change.historical.database 'options
        'add.cover.types)
   (put.value 'change.historical.database 'message
     (format ()
"No databases are currently loaded.  Select a database to load:-"
        (show-available-databases))))

(defun show-available-databases ()
"Updates the interface to show the available databases."
   (let ((dbs (list-available-databases)))
      (cond ((null dbs) (put.value 'change.historical.database 'message
                        "No databases are available"))
           (t (remove.all.values 'change.historical.database
                      'available.databases)
              (put.facet.value 'change.historical.database 'available.databases
                  'valueclass (cons 'one.of dbs))
              (unitmsg
                 'windowpane-available.databases-of-change.historical.database.1
                 'update!)
              (unitmsg
                 'windowpane-available.databases-of-change.historical.database.1
                 'open!)))))

(defun list-available-databases ()
"Returns a list of all the files in the subdirectory historical-data."
   (let ((dbs nil))
      (dolist (item (directory "historical-data/*") dbs)
         (let ((file (file-namestring item)))
            (unless (equal file "")
              (push file dbs))))))

(defun show-loaded-databases ()
"Updates the interface to show the available databases"
   (let ((dbs (get.values 'change.historical.database 'loaded.databases)))
      (cond ((null dbs) (put.value 'change.historical.database 'message
                        "No databases are currently loaded"))
           (t (remove.all.values 'change.historical.database
                      'database.to.remove)
              (put.facet.value 'change.historical.database 'database.to.remove
                  'valueclass (cons 'one.of (cons 'all dbs)))
```

JJM SYSTEMS INC

```
(unitmsg
    'windowpane-database.to.remove-of-change.historical.database.2
    'update!)
(unitmsg
    'windowpane-database.to.remove-of-change.historical.database.2
    'open!)))))

(defun remove-historical-database (db)
"Remove all the historical database units originating from the named database
from VEG."
    (remove.value 'change.historical.database 'loaded.databases db)
    (dolist (cover-type (unit.children 'historical.cover.types 'subclass))
      (when (equal db (get.value cover-type 'database))
        (dolist (sun-ang (unit.children cover-type 'subclass))
            (dolist (wave (unit.children sun-ang 'member))
              (delete.unit wave))
            (delete.unit sun-ang))
        (delete.unit cover-type)))
    (update-current-cover-types))

(defun remove-all-historical-database ()
"Remove all the historical database units from VEG."
    (remove.all.values 'change.historical.database 'loaded.databases)
    (dolist (cover-type (unit.children 'historical.cover.types 'subclass))
      (dolist (sun-ang (unit.children cover-type 'subclass))
        (dolist (wave (unit.children sun-ang 'member))
            (delete.unit wave))
        (delete.unit sun-ang))
      (delete.unit cover-type))
    (update-current-cover-types))

(defun update-data-matcher-if-necessary ()
"If the see descriptions screen of the data matcher in the browser is currently
open, update this screen."
    (when (eq (get.value 'viewport-data.matcher.2 'openp) 'open)
      (update-browser-see-descriptions-screen)))

(defun update-current-cover-types ()
"Updates the current cover types slot of the unit historical cover types."
    (put.value 'historical.cover.types 'current.cover.types
      (get-unit-names (unit.children 'historical.cover.types 'subclass))))

(defun update-pick-subset-screen ()
"Updates the pick subset screen for selecting restricted historical data."
    (remove.all.values '3.create.restricted.data 'ct.to.use)
    (put.facet.value '3.create.restricted.data 'ct.to.use 'valueclass
        (cons 'one.of (get.value 'historical.cover.types
                          'current.cover.types)))
    (unitmsg
      'windowpane-ct.to.use-of-3.create.restricted.data.3
      'update!))
```

JJM
SYSTEMS INC

```
(defun update-browser-see-descriptions-screen ()
"Updates the see descriptions screen from the browser."
(let ((cts (get.value 'historical.cover.types
                      'current.cover.types))
      (current-ct (get.value 'data.matcher 'ct.to.use)))
  (remove.all.values 'data.matcher 'ct.to.use)
  (put.value 'data.matcher 'ct.full.description "")
  (cond (cts
          (put.facet.value 'data.matcher 'ct.to.use 'valueclass
            (cons 'one.of cts))
          (unitmsg
           'windowpane-ct.to.use-of-data.matcher.4
           'update!)
          (unitmsg
           'windowpane-ct.to.use-of-data.matcher.4 'open!)
          (when (unit.exists.p current-ct)
            (put.value 'data.matcher 'ct.to.use current-ct)))
        (t (put.value 'data.matcher 'ct.full.description
              "No cover types are currently available")
          (unitmsg
           'windowpane-ct.to.use-of-data.matcher.4 'close!)))))
```

NASA National Aeronautics and Space Administration	Report Documentation Page		
1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle An Expert System Shell for Inferring Vegetation Characteristics - Changes to the Historical Cover Type Database (Task F)		5. Report Date May 1993	
		6. Performing Organization Code	
7. Author(s) P. Ann Harrison and Patrick R. Harrison		8. Performing Organization Report No. C931020-U-2R05	
		10. Work Unit No. 462-61-14	
9. Performing Organization Name and Address JJM Systems, Inc. One Ivybrook Blvd., Suite 190 Ivyland, PA 18974		11. Contract or Grant No. NAS5-30127	
		13. Type of Report and Period Covered Task Report for Task F March - May 1993	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546-0001 NASA/Goddard Space Flight Center Greenbelt, MD 20771			
		14. Sponsoring Agency Code	

15. Supplementary Notes

The Lisp and KEE code for this work is available on a Sun Cartridge Tape.

16. Abstract

The NASA VEGetation Workbench (VEG) is a knowledge based system that infers vegetation characteristics from reflectance data. All the options in VEG make use of a database of historical cover types. This database contains results from experiments by scientists on a wide variety of different cover types. In the previous version of VEG, the historical cover type database was stored as part of the VEG knowledge base. This database has been removed from the knowledge base. It is now stored as a series of flat files that are external to VEG. The report summarizes the use of the historical cover type database in VEG. It then describes the new interface to the files containing the historical data. Runs to test the operation of the new interface and to test the operation of VEG using historical data loaded from external files are also described.

17. Key Words (Suggested by Author(s)) EXPERT SYSTEM, ARTIFICIAL INTELLIGENCE, REMOTE SENSING, LEARNING, DISCRIMINATION	18. Distribution Statement UNCLASSIFIED - UNLIMITED		
19. Security Classif. (of this report) UNCLASSIFIED	20. Security Classif. (of this page) UNCLASSIFIED	21. No. of pages 37	22. Price

NASA FORM 1626 OCT 86

For sale by the National Technical Information Service, Springfield, VA 22161-2171